

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (Currently Amended) A system of managing security for data processing applications, comprising:

directories in which the data processing applications are stored, said directories being organized in an n-level tree; and

a ~~number~~ plurality of security registers which are selectively allocatable to any one of a plurality of said directories in response to the granting of rights in the directories and, after having been allocated to one directory, de-allocatable from said one directory and allocatable to another directory in response to granting of rights in said other directory, wherein each security register ~~containing~~ contains all rights or secrets which have been granted under the directory to which it has been allocated.

2. (Currently Amended) A method of managing security for data processing applications, comprising the steps of:

~~dynamically~~ allocating a security register to one of a plurality of directories that are organized in an n-level hierarchy, in response to granting of rights in said one directory;

storing in said allocated security register rights granted under the directory to which said security register has been allocated, ~~according to given rules;~~

seeking secrets presented in the directory in which a data processing application is stored; [[and]]

verifying knowledge of one or more rights at the level of said data processing application; and

in response to granting of rights in another directory, de-allocating said security register from said one directory and allocating said security register to said other directory.

3. (Currently Amended) A method ~~according to claim 2, wherein said allocating step is performed~~ of managing security for data processing applications, comprising the steps of:

dynamically allocating a security register to one of a plurality of directories that are organized in an n-level hierarchy according to the following rules:

allocation of a security register to a current directory as soon as a right has been granted under this directory or said security register has been updated if a right has already been granted under this directory;

removing the allocation of a security register to a current directory when a new directory is selected except if the selected new directory is a child of the current directory; and

allocating the security register that was allocated earliest to said new directory if the security registers are all allocated;

storing in said allocated security register rights granted under the directory to which said security register has been allocated, according to given rules;

seeking secrets presented in the directory in which a data processing application is stored; and

verifying knowledge of one or more rights at the level of said data processing application.

4. (Currently Amended) A method according to claim 2 wherein said seeking step is performed according to the following rule:

verifying that a secret presented is known in a current directory [[Ni]] or in a parent directory of said current directory at a higher level of the hierarchy.

5. (Currently Amended) A method ~~according to claim 2, wherein said seeking step comprises the following intermediate steps:~~ of managing security for data processing applications, comprising the steps of:

dynamically allocating a security register to one of a plurality of directories that are organized in an n-level hierarchy;

storing in said allocated security register rights granted under the directory to which said security register has been allocated, according to given rules;

seeking a secret in [[the]] a current directory at level (Ni) in which a data processing application is stored, and verifying the existence of [[the]] a secret within the application;

if said secret exists, verifying that presentation of the secret has succeeded;

if the presentation has succeeded, granting the right associated with the secret at the level (N_i) of the current application;

if the presentation has failed, refusing to grant the right associated with the secret and terminating the attempted presentation;

if said secret does not exist within the current application at level (N_i), determining whether ~~[[this]]~~ said secret exists within the parent application at level $N(i-1)$;

if said secret exists in the parent application at level $N(i-1)$, verifying that the presentation has succeeded;

if the presentation has succeeded, granting the right associated with the secret in the current application at level (N_i);

if the presentation has failed, refusing to grant the right associated with the secret and terminating the attempted presentation;

if the secret does not exist within the parent application at level $N(i-1)$, seeking the existence of the secret at the level of the application at level $N(i-2)$ within the hierarchy and verifying that the presentation has succeeded;

and so on as far as the highest hierarchical level as long as the existence of the secret has not been discovered;

if the secret has not been discovered, terminating the attempted presentation.

6. (Currently Amended) A method ~~according to claim 2, wherein the~~
~~verifying step is performed~~ of managing security for data processing applications,
comprising the steps of:

dynamically allocating a security register to one of a plurality of directories that are organized in an n-level hierarchy;

storing in said allocated security register rights granted under the directory to which said security register has been allocated, according to given rules;

seeking secrets presented in the directory in which a data processing application is stored; and

verifying knowledge of one or more rights at the level of said data processing application, according to the following rule:

authorization of a function requiring knowledge of a secret if and only if, within the hierarchy from the current application to the root application, a first secret is known to at least one of the applications along a path in the hierarchy for which the current application and the application containing the secret are delimiters.

7. (Currently Amended) A method ~~according to claim 2, wherein said verifying step comprises the following steps:~~ of managing security for data processing applications, comprising the steps of:

dynamically allocating a security register to one of a plurality of directories that are organized in an n-level hierarchy;

storing in said allocated security register rights granted under the directory to which said security register has been allocated, according to given rules;

seeking secrets presented in the directory in which a data processing application is stored;

verifying that a security register is associated with a current application at level N_i ;

authorizing a function if the security register contains a required right and terminating the verification;

seeking the existence of a reference secret within the current application at level N_i if no security register is associated with the current application or if the associated register does not contain the required right;

refusing the function and terminating the verification if the secret exists within the current application;

verifying that a security register is associated with the parent application at level $N(i-1)$ of the current application if the reference secret does not exist within the current application at level N_i ;

authorizing the function and terminating the verification if the security register associated with the parent application contains the right required for using the function;

seeking the existence of the reference secret within the parent application at level $N(i-1)$ of the current application if no security register is associated with the parent application or if the associated security register does not contain the required right;

refusing the function and terminating the verification if the reference secret exists within the parent application at level $N(i-1)$;

verifying that a security register is associated with the grandparent application at level $N(i-2)$ of the current application along a path within the hierarchy from the

current application towards the root application, if the reference secret does not exist within the parent application at level $N(i-1)$;

and so on as long as the existence of the reference secret has not been discovered; and

refusing the function and terminating the verification if the secret has not been discovered.